Some Use Cases for Distributed Computing Cafe Group

Greg Graham

Fermilab CD/CMS

4 May 2001

Introduction

- Focus on use cases for production systems
- Introduce some other use cases for distributed analysis
- Description of CMS production systems
  - In general
  - What we have at FNAL
- Conclusions - Towards Requirements and Scenarios

Some Use Cases

- The physicist specifies a job for complete production processing; he specifies parameters for generation, simulation, digitization, ntuple creation, and CMS software version.
  - Could be a mixture of physicists and production experts at first; eventually I see this as a tool for physicists.
  - I am implicitly assuming that the production job could run anywhere.

Some Use Cases

- The physicist specifies a job for partial production processing; he specifies the input data set and parameters starting at digitization/pileup stage for example, for software versions of existing processed data.
  - Once again, I am implicitly assuming that the input data and the processing may not reside/take place at the local site.
  - Important point is the specification of data.

Some Use Cases

- ***The physicist tracks progress of a production processing job.***
- ***The physicist queries for existing data that matches or nearly matches his search criterion. (Monte carlo or real data.)***
  - ***Exact match may not be necessary on things like minor software version number.***

Some Use Cases

- *The physicist specifies data to be run over with his own analysis program. The physicist may submit his own executable along with the job with a description of the output.*
  - *Once again, this may not happen at the local site.*
- *The physicist declares data to be of no more use.*
  - *This could be important; nature abhors empty tapes*

Production Systems - General

- ***Some Definitions:***
  - ***Runtime Environment*** - The collection of all environment variables, executables, and shared libraries needed to accomplish some task but not specifying the task.
  - **Job Environment** - The collection of all environment variables, executables, and shared libraries needed to specify some task.
- ***I'm sure that there are exceptions to some of the***

Production Systems - General

- There is a valid CMS runtime environment accessible from all production nodes.
- There is an Objectivity federation present with AMS server and Lock server accessible from all production nodes.
- ***There is some mass storage repository accessible to serve fz files and store results, log files, etc.***
- ***Production proceeds in a vertical fashion (ie- al***

Production Systems - General

- ***Production is script driven; specification/documentation of the production process is in the scripts.***
- ***Validation is log-file driven.***
- ***Tracking is script driven and not database driven.***

FNAL System

- cms_prod_util, v1_0_2, available from FNAL CD cvs server.
- CMSIM scripts not yet released ...
- Self tracking; if components are run automatically at intervals then complete OOHIT/OODigi processing is initiated merely by appearance of new fz files.
- CARF run number information is kept and cross correlated with the fz files and other CARF run numbers and the batch job ids. This will help validation

Conclusion - Towards Requirements and Scenarios

- Location Transparency:
  - Functional independence of where the operation is carried out and where the data resides.
- Robustness:
  - Robust against single points of failure
- Scalability:
  - Robust against addition of users/institutes and resources
- Optimizable